# Sadržaj

**1. Uvod ..... 1**

**2. Podatci ..... 2**

2.1. Odabir konferencija ..... 2

2.2. Pohrana podataka ..... 4

**3. Procesiranje PDF radova ..... 5**

3.1. Format znanstvenih radova ..... 6

3.2. Odabir parsera ..... 8

3.2.1. pdfminer ..... 8

3.2.2. pdfplumber ..... 8

3.2.3. PyPDF2 ..... 11

**4. Programska implementacija rješenja ..... 13**

4.1. Izvlačenje nerecenziranih izvora ..... 14

4.2. Pretraživanje referenci u radovima ..... 16

4.3. Analiza konteksta citiranja ..... 18

**5. Rezultati ..... 19**

5.1. Korišteni nerecenzirani izvori ..... 20

5.2. Kontekst korištenja nerecenziranih izvora ..... 22

**6. Zaključak ..... 25**

**Literatura ..... 26**

# 1. Uvod

Posljednjih godina, kibernetička sigurnost postala je izuzetno važna tema zbog rastućeg broja računalnih napada i incidenata [1]. Kako bi mogli razumjeti takvo složeno i promjenjivo područje, razni istraživači su se oslanjali na različite izvore informacija kako bi prikupili podatke o takvim događajima. Međutim, kao što je dobro poznato, prve informacije o napadima i incidentima u kibernetičkom prostoru obično se pojavljuju na raznim novinskim portalima, blogovima ili društvenim mrežama kao što su Facebook ili Twitter. Tek nakon određenog vremenskog razdoblja, informacije se šire u znanstvene izvore. Upravo iz tog razloga, dobar dio javno dostupnih podataka o računalnim napadima i incidentima potječe iz neprovjerenih i nerecenziranih izvora. Često se kao izvor informacija koriste upravo novinski članci, izvješća, sudski postupci i blogovi koji opisuju specifične događaje. Glavne karakteristike većine dostupnih izvora podataka su nedostatak recenziranja od strane stručnjaka i nedostupnost povjerljivih podataka.

S obzirom na činjenicu da je vrlo teško ručno pregledati sve moguće izvore i ocijeniti njihovu kvalitetu, cilj ovog rada bit će automatizirati identifikaciju pouzdanih izvora koji se koriste u znanstvenim radovima. Kako bismo identificirali nerecenzirane izvore iz područja kibernetičke sigurnosti, analizirat ćemo reference na te izvore u najbolje ocijenjenim radovima s vodećih svjetskih konferencija. U okviru ovog istraživanja analizirali smo 530 članaka s najpoznatijih svjetskih konferencija iz područja računalne sigurnosti. Također, proučavamo na koji način istraživači koriste nerecenzirane izvore – samo kao motivaciju za rad u uvodnom dijelu članka ili možda kao podršku argumenata u razradi i glavnom dijelu istraživanja.

Rad je strukturiran na sljedeći način, u drugom poglavlju rada opisan je način na koji su izabrane referentne konferencije. Treće poglavlje posvećeno je mogućnostima programskog jezika Python za izvlačenje i manipulaciju teksta. Prikazano je ponašanje različitih alata i tehnika koje se koriste za ovakve probleme i opisan je najčešći format znanstvenih radova. Zatim, u četvrtom poglavlju rada, opisana je programska implementacija rješenja i objašnjeni su korišteni regularni izrazi. U petom poglavlju rada, analizirani su rezultati te je u šestom poglavlju iznesen zaključak.

# 2. Podatci

S obzirom na očiglednu činjenicu da nije moguće ručno analizirati baš svaki nerecenzirani izvor, bilo je potrebno razviti metodologiju za analizu tih izvora na neki drugi način. Odabran je skup izvora podataka kojima možemo vjerovati i analizirano je koji nerecenzirani izvori se koriste u tim radovima.

Za analizu su odabrani radovi s najboljih i najuglednijih konferencija iz područja računalne sigurnosti. Naravno da kako raste ugled konferencije, tako raste i kontrola kvalitete sadržaja koji se koristi na spomenutim konferencijama. Najbolje ocijenjene konferencije imaju značajno stroža pravila kada pričamo o kontroli kvalitete korištenih izvora. Samim time, ako se za analizu koriste najuglednije konferencije u računalnoj sigurnosti, radovi moraju referencirati visoko kvalitetne izvore.

## 2.1. Odabir konferencija

Za odabir najuglednijih konferencija iz područja računalne sigurnosti, korištena je Microsoft Academic rang lista [2]. Među 10 najboljih konferencija s liste, neke konferencije nude opciju pristupa pojedinačnim radovima, dok druge konferencije objavljuju radove u obliku knjiga koje sadrže veliki broj pojedinačnih radova.

Ovakav primjer nekonzistentnosti u načinu objave radova među različitim konferencijama zakomplicirat će potrebnu automatizaciju jer je teško automatizirati analizu različito strukturiranih radova. Upravo zbog tog, odabrat ćemo 6 od 10 najuglednijih konferencija iz 2020. godine koje će se onda koristiti za daljnju analizu. Sve odabrane konferencije radove objavljuju pojedinačno, a spomenuti popis prikazan je u tablici Tablica 2.1.

Tablica 2.1: Najuglednije konferencije iz područja računalne sigurnosti u 2020. godini prema Microsoft Academic istraživanju

| **Rang** | **Konferencija** |
|---|---|
| 1 | IEE Symposium on Security and Privacy (S&P) 2020 |
| 2 | CCS'20 ACM SIGSAC Conference on Computer and Communications Security |
| 4 | SIGCOMM '20: Annual conference of the ACM – Special Interest Group on Data Communication on the applications, technologies, architectures and protocols for computer communication |
| 6 | Network and Distributed System Security Symposium 2020 (NDSS) |
| 8 | Annual Computer Security Applications Conference (ACSAC) 2020 |
| 9 | IEEE International Conference on Computer Communications (ICC) 2020 |

Ostale konferencije s popisa izbačene su iz daljnje analize zbog objavljivanja radova u obliku knjiga s velikim brojem pojedinačnih radova što bi izazvalo probleme kao što je spomenuto na prethodnoj stranici. Nakon što su izabrane konferencije koje će se koristiti u daljnjem radu, stvoren je skup podataka s radovima navedenih konferencija koji sadrži 530 radova.

## 2.2. Pohrana podataka

Kako bismo znanstvene članke mogli dalje analizirati, potrebno ih je dohvatiti iz određenog repozitorija. Spomenuti podatci nalaze se u mapama na Google Disku (engl. *Google Drive*). Google Disk je sustav za pohranu podataka u oblaku (engl. *cloud*). Jedna od ključnih značajki Google Diska je upravo mogućnost sinkronizacije podataka između oblaka i lokalne pohrane na računalu kroz Google Disk računalnu aplikaciju. Taj postupak često se naziva mapiranje Google Disk mapa lokalno.

Instalacija Google Disk aplikacije je vrlo jednostavan proces, potrebno je samo posjetiti Google Disk stranicu za preuzimanje, odabrati verziju za odgovarajući operacijski sustav i slijediti upute na zaslonu. Nakon što je instalacija završena, potrebno se samo prijaviti na Google račun kako bi se aplikacija povezala.

Za mapiranje mapa lokalno, Google Disk aplikacija nudi dvije glavne opcije:

- Zrcaljenje datoteka
    - Zrcaljenje datoteka preuzet će datoteke s Google Diska lokalno, stvarajući zrcalnu kopiju. Bilo kakve promjene načinjene na datotekama lokalno će se odraziti u oblaku i obrnuto, omogućujući sinkronizaciju u stvarnom vremenu. Međutim, ova opcija zahtijeva količinu lokalne pohrane ekvivalentnu datotekama na Google Disku.
- Prijenos datoteka
    - S druge strane, prijenos datoteka omogućava pregled i organizaciju datoteka na Google Disku s računala bez korištenja lokalne pohrane. Datoteke se prikazuju na računalu kao da su pohranjene lokalno, ali ostaju u oblaku dok im se ne treba pristupiti. Kada se određena datoteka otvori, preuzima se lokalno.

# 3. Procesiranje PDF radova

PDF format danas je jedan od najpopularnijih i najraširenijih formata koji se koristi u različitim industrijama širom svijeta. Kao rezultat toga, ogromne količine PDF datoteka generiraju se svakim danom što znači da imamo velike količine dostupnih informacija koje je potom potrebno analizirati i obraditi. Postoje razne tehnike i alati za obradu i analizu PDF dokumenata.

Jedan od najpogodnijih programskih jezika za takve zadatke je Python koji je zasigurno među najpopularnijim programskim jezicima današnjice, posebno kada govorimo o analizi i obradi podataka. Budući da je iznimno intuitivan i jednostavan jezik u usporedbi s drugim jezicima, potrebno je manje vremena i truda za izvođenje određenih zadataka. Sintaksa i čitljivost čine ga učinkovitim, lako obradivim i jednostavnim za učenje. Python već sadrži ugrađene funkcije za obradu znakovnih nizova, regularnih izraza i druge module za obradu teksta koji omogućuju normalizaciju, filtriranje i analizu izvučenog teksta. Također nudi široki raspon biblioteka i alata koji olakšavaju izvlačenje i manipulaciju teksta, meta podataka i ostalih informacija sadržanih u PDF dokumentima. I kao još jedna važna stavku kod odabira Pythona, mora se navesti velika i aktivna zajednica programera. Može se pronaći obilje resursa, uputa i primjera za obradu i analizu PDF dokumenata u Pythonu. Ako se naiđe na probleme ili nedoumice, velike su šanse da je netko već imao sličan problem i može pružiti pomoć s rješavanjem problema.

Sve navedene prednosti čine Python idealnim izborom za izgradnju programa za obradu velikih količina podataka.

## 3.1. Format znanstvenih radova

Znanstveni radovi koji se objavljuju u časopisima ili na znanstvenim konferencija često su formatirani u dva stupca. Primjer rada formatiranog na navedeni način prikazan je na (Slika 3.1). Takav format koristi se kako bi se optimizirao prostor na stranici i omogućila bolja čitljivost. Dva stupca omogućavaju autorima da na istu stranica smjeste više teksta, slika i tablica, čime se omogućava veća količina informacija na ograničenom prostoru.

Ovaj format obično ima ravnotežu između dužine redaka i visine stupaca kako bi se osigurala lakša čitljivost. Kraći redovi smanjuju potrebu za premještanjem očiju čitatelja s kraja jednog retka na početak sljedećeg. S druge strane, dva stupca pružaju vizualan okvir za prikaz paralelnih informacija ili komparativnih analiza. Format s dva stupca posebno je koristan u znanstvenim časopisima koji imaju stroga pravila o maksimalnom broju stranica po radu, iako je bitno napomenuti da se ne radi o odabiru između jednog ili dva stupca.

Sve u svemu, format znanstvenih radova s dva stupca omogućuje ekonomično korištenje prostora i efikasan prikaz informacija i zbog tih prednosti, široko se koristi u znanstvenoj zajednici zbog svoje praktičnosti i estetske privlačnosti.

U skupu podataka nad kojim ćemo provoditi analizu, velika većina radova je upravo u ovom formatu i zato je bitno da kod odabira alata za izvlačenje teksta nađemo prikladan alat koji će najbolje prepoznati stupce. Kao što ćemo vidjeti u sljedećem poglavlju, neki alati nažalost neće dobro prepoznati stupce i u cijelosti će u radovima čitati retke od početka do kraja bez prepoznavanja stupaca.

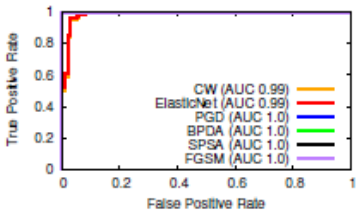


Figure 7: ROC Curve of detection on MNIST with single-label defense.

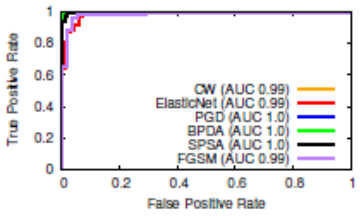


Figure 8: ROC Curve of detection on CIFAR10 with single-label defense.

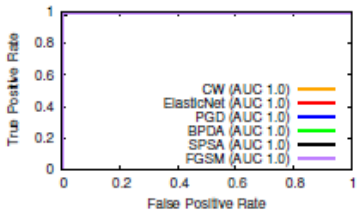


Figure 9: ROC Curve of detection on YouTube Face with single-label defense.

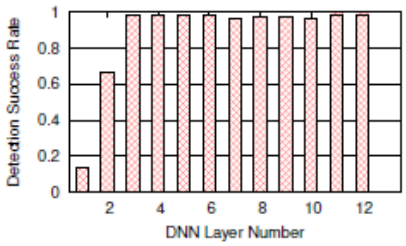


Figure 10: Detection success rate of CW attack at 5% FPR when using different layers for detection in a GTSRB model.

Table 8: ResNet20 Model Architecture for CIFAR10.

| Layer Name (type) | # of Channels | Activation | Connected to |
|---|---|---|---|
| conv_1 (Conv) | 16 | ReLU | - |
| conv_2 (Conv) | 16 | ReLU | conv_1 |
| conv_3 (Conv) | 16 | ReLU | pool_2 |
| conv_4 (Conv) | 16 | ReLU | conv_3 |
| conv_5 (Conv) | 16 | ReLU | conv_4 |
| conv_6 (Conv) | 16 | ReLU | conv_5 |
| conv_7 (Conv) | 16 | ReLU | conv_6 |
| conv_8 (Conv) | 32 | ReLU | conv_7 |
| conv_9 (Conv) | 32 | ReLU | conv_8 |
| conv_10 (Conv) | 32 | ReLU | conv_9 |
| conv_11 (Conv) | 32 | ReLU | conv_10 |
| conv_12 (Conv) | 32 | ReLU | conv_11 |
| conv_13 (Conv) | 32 | ReLU | conv_12 |
| conv_14 (Conv) | 32 | ReLU | conv_13 |
| conv_15 (Conv) | 64 | ReLU | conv_14 |
| conv_16 (Conv) | 64 | ReLU | conv_15 |
| conv_17 (Conv) | 64 | ReLU | conv_16 |
| conv_18 (Conv) | 64 | ReLU | conv_17 |
| conv_19 (Conv) | 64 | ReLU | conv_18 |
| conv_20 (Conv) | 64 | ReLU | conv_19 |
| conv_21 (Conv) | 64 | ReLU | conv_20 |
| pool_1 (AvgPool) | - | - | conv_21 |
| dropout_1 (Dropout) | - | - | pool_1 |
| fc_ (FC) | - | Softmax | dropout_1 |

- *Traffic Sign Recognition* (GTSRB) – Here the goal is to recognize 43 different traffic signs, emulating an application for self-driving cars. We use the German Traffic Sign Benchmark dataset (GTSRB), which contains 35.3K colored training images and 12.6K testing images [44]. The model consists of 6 convolution layers and 2 dense layers (see Table 7). This task is 1) commonly used as an adversarial defense evaluation benchmark and 2) represents a real-world setting relevant to our defense.
- *Image Recognition* (CIFAR10) – The task is to recognize 10 different objects. The dataset contains 50K colored training images and 10K testing images [23]. The model is an Residual Neural Network (RNN) with 20 residual blocks and 1 dense layer [20] (Table 8). We include this task because of its prevalence in general image classification and adversarial defense literature.
- *Face Recognition* (YouTube Face) – This task is to recognize faces of 1,283 different people drawn from the YouTube videos [51]. We build the dataset from [51] to include 1,283 labels, 375.6K training images, and 64.2K testing images [13]. We use a large ResNet-50 architecture architecture [20] with over 25 million parameters. We include this task because it simulates a more complex facial recognition-based security screening scenario. Defending against adversarial attack in this setting is important. Furthermore, the large set of labels in this task allows us to explore the scalability of our trapdoor-enabled detection.

**Model Architecture.** We now present the architecture of DNN models used in our work.

- **MNIST** (Table 6) is a convolutional neural network (CNN) consisting of two pairs of convolutional layers connected by max pooling layers, followed by two fully connected layers.
- **GTSRB** (Table 7) is a CNN consisting of three pairs of convolutional layers connected by max pooling layers, followed by two fully connected layers.
- **CIFAR10** (Table 8) is also a CNN but includes 21 sequential convolutional layers, followed by pooling, dropout, and fully connected layers.
- **YouTube Face** is the ResNet-50 model trained on the YouTube Face dataset. It has 50 residual blocks with over 25 millions parameters.

**Detailed information on attack configuration.** We evaluate the trapdoor-enabled detection using six adversarial attacks: CW, ElasticNet, PGD, BPDA, SPSA, and FGSM (which we have described in Section 2.1). Details about the attack configuration are listed in Table 10.

**Sample Trapdoor Patterns.** Figure 11 shows sample images that contain a single-label defense trapdoor (a single $6 \times 6$ square) and that contain an all-label defense trapdoor (five $3 \times 3$ squares). The mask ratio of the trapdoors used in our experiments is fixed to $\kappa = 0.1$.

Slika 3.1: Primjer dvostrukog stupca u znanstvenim radovima [4]

## 3.2. Odabir parsera

PDF omogućuje ugradnju raznih vrsta multimedijskih sadržaja i privitaka, ali PDF parseri se pretežno koriste za izvlačenje tekstualnih paragrafa, jedinica podataka (datumi, identifikatori, brojevi), slika i tabličnih podataka.

U našem slučaju, fokusirat ćemo se na izvlačenje teksta iz PDF dokumenata.

Python nudi bogati ekosustav alata za rad s PDF dokumentima. Neki popularni paketi i biblioteke uključuju *pdfminer, pdfplumber* i *PyPDF2*.

Ove biblioteke omogućuju jednostavno izvlačenje teksta i ostalih informacija iz PDF dokumenata. Sve navedene biblioteke su isprobane i doneseni su određeni zaključci koji diktiraju daljnji tijek izrade parsera.

### 3.2.1. pdfminer

Jedan od najpopularnijih Python paketa za izvlačenje informacija iz PDF dokumenata je pdfminer [5]. Dokumentacija je oskudna, ali može se pronaći dosta odlomaka koda koji onda mogu pomoći kod implementacije rješenja.

S obzirom na to da je pdfplumber izgrađen na temelju pdfminer-a, detaljnije ćemo opisati pdfplumber koji bi po svemu sudeći trebao imati mnogo širi spektar mogućnosti.

### 3.2.2. pdfplumber

pdfplumber je Python biblioteka koja je izgrađena koristeći pdfminer i nudi niz mogućnosti za interakciju s PDF datotekama i podržava različite verzije Pythona kao što su Python 3.7, 3.8, 3.9 i 3.10. Dokumentacija pdfplumber-a je definitivno najkompletnije i najintuitivnije napisana među testiranim bibliotekama [6]. Valja napomenuti kako pdfplumber najboljem radi s generiranim PDF-ovima, dok dolazi do određenih problema kad se tekst izvlači iz skeniranih PDF-ova.

pdfplumber omogućava pristup i izvlačenje različitih tipova PDF objekata kao što su tekst, linije, slike, anotacije i hiperlinkovi. Treba imati na umu da je pdfplumber izgrađen na bazi pdfminer-a što znači da će bilo kakvi problemi ili ograničenja sa pdfminer-om potencijalno utjecati i na rad pdfplumber-a.

pdfplumber pruža različite klase, objekte i metode za rad s PDF dokumentima. U nastavku su opisane dvije najvažnije klase koje se koriste.

`pdfplumber.PDF` klasa predstavlja jedan PDF dokument i sadrži dva glavna svojstva koja su navedena u tablici Tablica 3.1.

Tablica 3.1: Svojstva `pdfplumber.PDF` klase

| **Svojstvo** | **Opis** |
|---|---|
| `.metadata` | Rječnik ključ/vrijednost parova, uglavnom sadrži *"CreationDate", "ModDate", "Producer"* |
| `.pages` | Lista koja sadrži jednu instancu `pdfplumber.page` klase po učitanoj stranici |

`pdfplumber.Page` klasa predstavlja jednu stranicu PDF dokumenta. Većina stvari za koje se koristi pdfplumber temeljit će se upravo na ovoj klasi. Navedena klasa sadrži sljedeća glavna svojstva koja su navedena u tablici Tablica 3.2.

Tablica 3.2: Svojstva `pdfplumber.Page` klase

| **Svojstvo** | **Opis** |
|---|---|
| `.extract_text()` | Metoda koja vraća znakovni niz u kojem je sadržan čitav tekst sa stranice |
| `.page_number` | Redni broj stranice, počevši od 1 za prvu stranicu |
| `.width` | Širina stranice |

| | |
|---|---|
| `.height` | Visina stranice |
| `.objects` | Lista svih objekata na trenutnoj stranici |
| `.chars` | Lista svih znakova na trenutnoj stranici |
| `.images` | Lista svih slika na trenutnoj stranici |
| `.annots` | Lista svih anotacija na trenutnoj stranici |
| `.hyperlinks` | Lista svih hiperveza na trenutnoj stranici |

Jednostavni primjer ispisivanja teksta stranicu po stranicu:

```
import pdfplumber

with pdfplumber.open("example.pdf") as pdf:
    for page in pdf.pages:
        text = page.extract_text()
        print(text)
```

Kôd 3.1 – Odsječak koda za ispis svih stranica PDF dokumenta koristeći pdfplumber

Jedina i presudna mana pdfplumber i pdfminer biblioteke je loše automatsko prepoznavanje rasporeda teksta i ostalih PDF objekata pa iz tog razloga često dolazi do situacije kada pdfplumber ne prepozna da se radi o PDF dokumentu u kojem su stranice podijeljene u dva stupca. U tom slučaju treba čitati stupac po stupac, međutim dolazi do problema jer se čita cijeli prvi red i onda je izvučeni tekst poprilično beskoristan i nema nikakvu logičku povezanost. Ovakav problem može se riješiti analizom rasporeda (engl. *layout analysis*), ali to je problem za sebe i zato ćemo koristiti PyPDF2 koji je opisan u sljedećem poglavlju.

### 3.2.3. PyPDF2

PyPDF2 je Python biblioteka otvorenog koda bez ikakvih vanjskih zavisnosti za rad s PDF dokumentima. PyPDF2 omogućuje pretvorbu PDF dokumenata u slike ili tekstove, kreiranje novih PDF dokumenata i uređivanje postojećih PDF dokumenata dodavanjem, uklanjanjem i izmjenom sadržaja stranica dokumenta.

PyPDF2 sadrži klasu `PdfReader` koja je ekvivalentna klasi `pdfplumber.PDF` i predstavlja jedan PDF dokument te sadrži svojstva dosta slična prethodno spomenutima u klasi `pdfplumber.PDF`. Svojstva su opisana u tablici Tablica 3.3.

Tablica 3.3: Svojstva `PyPDF2.PdfReader` klase

| Svojstvo | Opis |
|---|---|
| `.metadata` | Rječnik ključ/vrijednost parova, uglavnom sadrži podatke kao što su *"Author", "Creator", "Producer", "Subject", "Title"* |
| `.pages` | Lista koja sadrži jednu instancu `PageObject` klase po učitanoj stranici |
| `.numPages` | Broj stranica PDF dokumenta |

`PageObject` klasa predstavlja jednu stranicu u PDF dokumentu. Navedena klasa će se tipično kreirati pozivom metode `get_page()` iz klase `PdfReader.PageObject` klasa sadrži sljedeća glavna svojstva koja su navedena u tablici Tablica 3.4.

Tablica 3.4: Svojstva `PyPDF2.PageObject` klase

| Svojstvo | Opis |
|---|---|
| `.extract_text` | Metoda koja vraća znakovni niz u kojem je sadržan čitav tekst sa stranice |
| `.images` | Lista svih slika na trenutnoj stranici |
| `.annotations` | Lista svih anotacija na trenutnoj stranici |

Iz sljedećeg primjera možemo vidjeti da je upotreba klase `PyPDF2.PdfReader` jako slična pdfplumber-u.

```
from PyPDF2 import PdfReader

reader = pdfReader("example.pdf")
for page in reader.pages:
   text = page.extract_text()
   print(text)
```

Kôd 3.2 – Primjer ispisa teksta svih stranica PDF dokumenta koristeći PyPDF2

PyPDF2 nam ne nudi nikakve prednosti u smislu performansi ili dodatnih mogućnosti u odnosu na pdfplumber, ali automatski ispravno prepoznaje stupce u PDF dokumentima i pravilno ih čita. Upravo ta razlika bila je odlučujuća kod odabira alata za daljnju analizu znanstvenih radova.

# 4. Programska implementacija rješenja

Prvi korak je identificirati često korištene izvore koji nisu prošli proces recenziranja. Početna pretpostavka je da se većina izvora referencira putem URL-a. Ova informacija koristi se kako bi izvukli sve URL-ove iz "*REFERENCES*" sekcije konferencijskih radova. Kako bi se izdvojili često korišteni izvore, URL-ovi će se grupirati prema njihovim domenskim imenima. Naravno, ovakav način pretrage neće obuhvatiti sve izvore jer nisu svi izvori dostupni na internetu.

Nerecenzirane konferencije često pružaju izravne informacije o kibernetičkoj sigurnosti. Tijekom analize, zamijećeno je da autori članaka na nekonzistentan način citiraju ove izvore, pri čemu neki autori navode URL, dok drugi navode ime konferencije i autore. Kako bismo u pretragu uključili i ovu vrstu izvora, koristili smo nekoliko popisa konferencija i pretražili naš skup podataka za spominjanje imena konferencija u odjeljcima s referencama i uključili ih na naš popis. Osim konferencija, skup podataka sadrži i ostale ugledne izvore koji bi se mogli koristiti kao izvori, ali se ne navode putem URL-a. Upravo zbog tog razloga, uz pretragu URL-ova, provest će se i pretraga skupa ključnih riječi koje bi mogle sadržavati takve izvore.

Nakon toga, zadatak je analizirati korištenje tih referenci, odnosno u kojem kontekstu se izvori citiraju u člancima. Članke ćemo raspodijeliti na uvod i pozadinu, razradu i istraživanje te raspravu i zaključak.

## 4.1. Izvlačenje nerecenziranih izvora

Postoje razni načini citiranja i referenciranja, ali početna pretpostavka je da se u sekciji referenci izvori referenciraju korištenjem IEEE preporuke, odnosno brojevima u uglatim zagradama. Pretraživanje "*REFERENCES*" sekcije implementirano je na način da se sprema broj reference iz uglatih zagrada i njemu se pridružuje domensko ime URL-a i/ili ključna riječ.

```
ref_list = re.findall(r'\[\d+\][^\[]*', ref_string)

for ref in ref_list:
    number = re.search(r'\[(\d+)\]', ref).group(1)

    url = re.findall(r'(https?://\S+)', ref)

    matching_keywords = [keyword for keyword in keywords
                         if keyword.lower() in ref.lower()]

    domain = ''
    if url:
        url = url[0].split()[0]
        domain = urlparse(url).netloc

    if number not in references.keys():
        references[number] = {'url': domain,
                              'keywords': matching_keywords}
```

Kôd 4.1 – Primjer ispisa teksta svih stranica PDF dokumenta koristeći pdfplumber

Opis regularnih izraza korištenih u Kôd 4.1:

- `ref_list = re.findall(r'\[\d+\][^\[]*', ref_string)`
  - ova linija koda koristi regularni izraz `r'\[\d+\][^\[]*'` za pronalaženje svih referenci u `ref_string`
  - `'\[\d+\]'`: traži tekst koji počinje s '[', zatim ima jedan ili više brojeva (`\d+`) i završava s ']'
  - `'[^\[]*'`: nakon toga, traži bilo koji znak koji nije '[', i to nula ili više puta. Dakle, ovaj red koda pronaći će sve znakovne nizove koji počinju s brojem u uglatim zagradama, i nastavljaju se sve do sljedeće uglate zagrade (ili do kraja znakovnog niza)

- `number = re.search(r'\[(\d+)\]', ref).group(1)`
    - ova linija koda koristi regularni izraz `r'\[(\d+)\]'` za pronalaženje broja unutar uglatih zagrada
    - metoda group(1) će vratiti prvu uhvaćenu grupu (engl. *capturing group*), tj. prvi dio teksta koji odgovara dijelu regularnog izraza unutar zagrada, odnosno ovo će izdvojiti broj reference
- `url = re.findall(r'(https?://\S+)', ref)`
    - ova linija koda koristi regularni izraz `r'(https?://\S+)'` za pronalaženje svih URL-ova u referenci
    - `'https?://'`: traži string koji počinje s "http://" ili "https://".
    - `'\S+'`: nakon toga, traži jedan ili više nerazmaknutih znakova, tj. ovaj red koda će izdvojiti sve URL-ove iz referenci

Osim regularnih izraza, važno je spomenuti i funkciju `urlparse` koja analizira URL i potom iz URL-a izdvoji domenu (`netloc`).

## 4.2. Pretraživanje referenci u radovima

Nakon što su identificirani redni brojevi svih referenci u određenom članku i rednim brojevima pridružene vrijednosti (domenska imena, ključne riječi), potrebno je evaluirati u kojem dijelu rada se citiraju pronađeni izvori.

Izvori se citiraju na jedan od sljedećih načina:

- "[1]"
    - predstavlja citiranje samo jednog izvora
    - najčešći oblik
- "[1, 2, 5]"
    - predstavlja skup od više izvora odvojenih zarezom
    - brojevi ne moraju biti sortirani uzlazno
- "[1 - 5] "
    - predstavlja raspon brojeva, ekvivalentno je obliku [1, 2, 3, 4, 5]
    - kad se citira više izvora odjednom, autori često koriste ovakav oblik za grupiranje referenci
    - problem kod pretraživanja ovakvog oblika je što će citirane vrijednosti biti skrivene u rasponu vrijednosti
- "[1 - 4, 7]"
    - moguće je i kombiniranje prethodno navedena 2 oblika
    - u ovom slučaju, to predstavlja oblik [1, 2, 3, 4, 7]

Prepoznavanje referenci citiranih u uglatim zagradama izvodit će se također regularnim izrazima. Nakon što su izvučeni sve brojevi koji se referenciraju u uglatoj zagradi, za svaki od tih brojeva dohvaća se vrijednost iz prethodno pohranjenog rječnika koji kao ključeve sadrži broj reference, a kao vrijednosti sadrži domenu i/ili ključnu riječ. Zatim se ažurira informacija o citiranom izvoru na način koji će biti opisan u poglavlju 4.3.

Sada ćemo opisati regularni izraz koji se koristi za prepoznavanje citiranih izvora, odnosno prepoznavanje uglatih zagrada koje odgovaraju formatu koji smo upravo opisali.

- `re.findall(r'\[\d+(?:\s*-\s*\d+)?(?:,\s*\d+)*\]', text)`
  - `'\[\d+'`: traži tekst koji počinje s "[", zatim ima jedan ili više brojeva (`\d+`)
  - `'(?:\s*-\s*\d+)?'`: traži crticu `(-)` nakon čega slijedi jedan ili više brojeva, to upravo predstavlja raspon, npr. "1-5" ili "1-3,5". Prije i nakon crtice se mogu nalaziti i opcionalni razmaci i to je obuhvaćeno sa `'s*'`. `(?:)` označava neuhvaćenu grupu (engl. *non-capturing group*), što znači da ova grupa neće biti vraćena kao rezultat pretrage. Također, ovaj dio regularnog izraza je neobavezan ('?' na kraju znači da dio može biti prisutan 0 ili 1 puta)
  - `'(?:,\s*\d+)*'`: ovo je još jedna neuhvaćena grupa koja odgovara zarezu ',' i zatim nula ili više razmaka '`\s*`' i jednoj ili više znamenaka. '`*`' na kraju će omogućiti nula ili više pojavljivanja zareza s opcionalnim razmacima, nakon čega slijedi jedna ili više znamenaka, a to omogućuje višestruke zarezom odvojene brojeve citata, poput "1, 2, 3" i slično.
  - `'\]'`: za kraj, ova linija koda odgovara zatvorenoj uglatoj zagradi "]"

Osim ukupnog broja pojava određenog izvora, za svaki izvor spremat će se lista datoteka u kojima se taj izvor pojavio, kao i broj različitih datoteka u kojima se taj rad pojavio.

Dakle, nakon analize svih radova, svakom izvoru bit će pridružen ukupan broj citiranja izvora, lista različitih radova u kojima je izvor citiran te broj različitih radova u kojima je izvor citiran.

## 4.3. Analiza konteksta citiranja

Primarni cilj ovog rada bio je pronaći koji trenutno korišteni nerecenzirani izvori postoje, a drugi cilj je procijeniti u kojem kontekstu autori koriste te izvore i je li upotreba izvora konzistentna s konferencije na konferenciju.

S obzirom na to da se različiti odjeljci radova razlikuju od rada do rada, teško je objektivno odlučiti koliki dio rada će sadržavati uvod, a koliki dio će sadržavati razradu teme ili zaključak. Prilikom dohvaćanja referenci kao što je opisano u poglavlju 4.2., spremat ćemo i podatak na kojoj stranici rada se izvor citira. Zatim ćemo broj stranice podijeliti s ukupnim brojem stranica tog rada i dobiti informaciju gdje je izvor citiran, izraženu kao postotak.

Podjela konteksta prikazana je u tablici Tablica 4.1.

Tablica 4.1: Podjela konteksta prema postotku udjela u radu

| **Postotak** | **Kontekst citiranja** |
|---|---|
| 0% - 30% | Uvod i pozadina – u ovom dijelu rada predstavlja se tema i kontekst istraživanja i pokazuje čitatelju koja je svrha rada |
| 30% - 75% | Razrada i istraživanje – u ovom dijelu rada opisuje se kako je istraživanje provedeno. To uključuje opisivanje metodologije, prikazivanje rezultate i analizu rezultata |
| 75% - 100% | Rasprava i zaključak – u posljednjem dijelu rada, raspravlja se o rezultatima i njihovim implikacijama, donose se zaključci i sugeriraju se smjernice za buduća istraživanja |

95% analiziranih radova ima minimalno 10 stranica pa bi ovakav pregled trebao dati zadovoljavajuće rezultate.

Nažalost, ne postoji jedinstveno rješenje za ovakav problem, jer neki radovi će možda zahtjevi puno dublji uvod u problem, dok će neki autori najveću pažnju posvetiti obradi rezultata.

# 5. Rezultati

Tijekom analize često korištenih izvora, uočeno je da su većina citiranih URL-ova zapravo veze do različitih identifikatora digitalnih objekata (DOI), različitih članaka ili repozitorija izvornog koda, umjesto da se radi o nerecenziranim izvorima.

Budući da navedeni tipovi izvora nisu relevantni za ovo istraživanje, iz daljnje analize isključili smo domene povezane sa sljedećim temama:

- DOI
- repozitoriji članaka i zbornici
- repozitoriji izvornog koda
- baze podataka o ranjivostima
- vijesti o različitim proizvodima i odgovarajućim ažuriranjima
- Wikipedia članci
- tehničke specifikacije
- vijesti o kriptovalutama
- dokumentacija o procesorima
- baze podataka o greškama

Tijekom pregleda rezultata istraživanja, primijećeno je nekoliko slučajeva u kojima je citirani izvor bila sigurnosna kompanija koja je u nekom trenutku bila preuzeta od strane druge tvrtke, što rezultira citiranjem imena obje kompanije u ovisnosti o tome kada je rad izvorno objavljen. Primjerice, budući da je Symantec preuzet od strane Broadcoma 2019. godine [8], članci se objavljuju pod drugim imenom - Broadcom, što rezultira citiranjem oba imena. Slično tome, Palo Alto Networks je preuzeo PureSec 2019. godine [9]. Nažalost ovakve probleme nije moguće automatizirati pa će se takvi slučajevi morati pregledati ručno.

Kako bismo suzili popis relevantnih izvora, odlučili smo analizirati samo one izvore koji su citirani pet ili više puta. Ako neki izvor ima frekvenciju manju od pet, smatramo ga rijetko korištenim i odbacujemo.

## 5.1. Korišteni nerecenzirani izvori

Analizom referenci korištenih na vodećim znanstvenim konferencijama iz područja računalne sigurnosti u 2020. godini, identificirali smo 15 izvora koji su bili citirani više od 5 puta. Ukupni broj citiranja kroz 530 radova sa 6 konferencija iznosi 139.

Ako se fokusiramo na vrstu izvora umjesto na broj citata pri pregledu rezultata u tablici Tablica 5.1, rezultati su sljedeći:

1) specijalizirane vijesti (33.3%)
2) nespecijalizirane, ali visoko ugledne vijesti (20%)
3) publikacije sigurnosnih kompanija (20%)
4) blogovi sigurnosnih kompanija (13.3%)
5) sve ostale vrste, poput konferencija, laboratorija, privatnih blogova i vijesti (13.3%)

Ako se umjesto toga fokusiramo na broj referenci koje potječu iz različitih vrsta izvora, rezultati su sljedeći:

1) specijalizirane vijesti (36%)
2) nespecijalizirane, ali visoko ugledne vijesti (23%)
3) publikacije sigurnosnih kompanija (22.3%)
4) blogovi sigurnosnih kompanija (10.8%)
5) sve ostale vrste, poput konferencija, laboratorija, privatnih blogova i vijesti (7.9%)

Zanimljivost je između ostalog, da su novine općeg sadržaja relevantni izvori informacija s 32 citiranja. Te novine su ugledni izvori pouzdanih vijesti. Naravno da opći izvori vijesti neće pokriti sve teme računalne sigurnosti, ali često pokrivaju važne događaje iz područja računalne sigurnosti i stoga se mogu smatrati valjanim izvorima informacija o određenom sigurnosnom problemu ili incidentu

Tablica 5.1: Nerecenzirani izvori korišteni u 6 analiziranih konferencija

| RANG | IZVOR | VRSTA IZVORA | BROJ CITIRANJA |
|---|---|---|---|
| **1** | Broadcom / Symantec | Sigurnosna kompanija | 16 |
| **2** | nytimes.com | Opće vijesti | 15 |
| **3** | zdnet.com | Specijalizirane vijesti | 14 |
| **4** | arstechnica.com | Specijalizirane vijesti | 11 |
| **5** | BleepingComputer.com | Specijalizirane vijesti | 10 |
| **6** | sophos.com | Sigurnosna kompanija | 10 |
| **7** | bbc.com | Opće vijesti | 9 |
| **8** | wired.com | Specijalizirane vijesti | 9 |
| **9** | theguardian.com | Opće vijesti | 8 |
| **10** | googleprojectzero.blogspot.com | Blog sigurnosne kompanije | 8 |
| **11** | security.googleblog.com | Blog sigurnosne kompanije | 7 |
| **12** | techcrunch.com | Specijalizirane vijesti | 6 |
| **13** | blackhat.com | Konferencija | 6 |
| **14** | Microsoft (Azure, MSDN) | Sigurnosna kompanija | 5 |
| **15** | citizenlab.ca | Laboratorij | 5 |
| | **UKUPNO** | | 139 |

## 5.2. Kontekst korištenja nerecenziranih izvora

Konačni cilj bio je saznati u kojim kontekstima autori koriste izvore koji nisu recenzirani, odnosno koriste li se ti izvori samo kako bi motivirali rad, pokazali da je određeno ponašanje pronađeno u praksi ili su ti podaci korišteni u glavnom dijelu istraživanja.

Rezultati ove analize prikazani su u tablici Tablica 5.2, koja prikazuje koje nerecenzirane izvore autori koriste, njihovu vrstu i način na koji se koriste u radovima.

Analizom korištenih referenci vidljivo je da se najveći dio izvora (55%) koristi za motivaciju, odnosno za uvod i predstavljanje pozadine rada. Manji dio izvora (37%) koristi se u razradi i istraživanju znanstvenih radova, dok se za raspravu i zaključak koristio najmanji dio (8%) nerecenziranih izvora.

Kao što je prikazano u tablici Tablica 5.3, neki izvori češće se koriste za samu motivaciju, dok drugi sadrže informacije koje se češće koriste u glavnom dijelu istraživanja. Ovi izvori su prihvaćeni na vodećim konferencijama iz područja računalne sigurnosti i stoga bi se i ostali istraživači mogli osloniti na njih jer objavljuju pouzdane informacije.

Tablica 5.2: Broj citiranja nerecenziranih izvora prema dijelu rada u kojem su citirani

| RANG | IZVOR | VRSTA IZVORA | UVOD I POZADINA | RAZRADA I ISTRAŽIVANJE | RASPRAVA I ZAKLJUČAK | BROJ CITIRANJA |
|---|---|---|---|---|---|---|
| **1** | Broadcom / Symantec | Sigurnosna kompanija | 8 | 7 | 1 | 16 |
| **2** | nytimes.com | Opće vijesti | 12 | 3 | 0 | 15 |
| **3** | zdnet.com | Specijalizirane vijesti | 11 | 3 | 0 | 14 |
| **4** | arstechnica.com | Specijalizirane vijesti | 8 | 3 | 0 | 11 |
| **5** | BleepingComputer.com | Specijalizirane vijesti | 6 | 3 | 1 | 10 |
| **6** | sophos.com | Sigurnosna kompanija | 0 | 10 | 0 | 10 |
| **7** | bbc.com | Opće vijesti | 3 | 5 | 1 | 9 |
| **8** | wired.com | Specijalizirane vijesti | 6 | 2 | 1 | 9 |
| **9** | theguardian.com | Opće vijesti | 2 | 6 | 0 | 8 |
| **10** | googleprojectzero.blogspot.com | Blog sigurnosne kompanije | 6 | 2 | 0 | 8 |
| **11** | security.googleblog.com | Blog sigurnosne kompanije | 4 | 1 | 2 | 7 |
| **12** | techcrunch.com | Specijalizirane vijesti | 5 | 1 | 0 | 6 |
| **13** | blackhat.com | Konferencija | 4 | 1 | 1 | 6 |
| **14** | Microsoft (Azure, MSDN) | Sigurnosna kompanija | 2 | 2 | 1 | 5 |
| **15** | citizenlab.ca | Laboratorij | 0 | 2 | 3 | 5 |

Tablica 5.3: Izvori s najvećim brojem citiranja s obzirom na dio rada u kojem se citiraju

| RANG | UVOD I POZADINA | BR. | RAZRADA I ISTRAŽIVANJE | BR. | RASPRAVA I ZAKLJUČAK | BR. |
|---|---|---|---|---|---|---|
| **1** | nytimes.com | 12 | sophos.com | 10 | citizenlab.ca | 3 |
| **2** | zdnet.com | 11 | Broadcom / Symantec | 7 | security.googleblog.com | 2 |
| **3** | Broadcom / Symantec | 8 | theguardian.com | 6 | bbc.com | 1 |
| **4** | arstechnica.com | 8 | bbc.com | 5 | Broadcom / Symantec | 1 |
| **5** | wired.com | 6 | nytimes.com | 3 | BleepingComputer.com | 1 |
| **6** | BleepingComputer.com | 6 | zdnet.com | 3 | wired.com | 1 |
| **7** | googleprojectzero.blogspot.com | 6 | arstechnica.com | 3 | Microsoft (Azure, MSDN) | 1 |
| **8** | techcrunch.com | 5 | BleepingComputer.com | 3 | | |
| **9** | blackhat.com | 4 | wired.com | 2 | | |
| **10** | security.googleblog.com | 4 | googleprojectzero.blogspot.com | 2 | | |
| **11** | wired.com | 6 | citizenlab.ca | 2 | | |
| **12** | bbc.com | 3 | Microsoft (Azure, MSDN) | 2 | | |

# 6. Zaključak

U ovom radu razvijena je metodologija za identificiranje i analizu konteksta korištenja nerecenziranih izvora na konferencijama iz područja računalne sigurnosti. Analizirane su najbolje konferencije o računalnoj sigurnosti iz 2020. godine i identificirano je ukupno 15 različitih nerecenziranih izvora s 5 ili više citata na promatranim konferencijama.

Nerecenzirani izvori u najvećoj mjeri koriste se za motivaciju rada, odnosno autori ih koriste u uvodnom dijelu rada u 55% slučajeva. Nakon toga slijedi dio razrade teme i glavnog istraživanja s 37%, dok ostalih 8% pada u dio rasprave i zaključka. Istraživanja sugeriraju da različiti izvori dominiraju u različitim dijelovima članaka, vjerojatno jer ne pružaju istu vrstu informacija.

Većina izvora su publikacije i blogovi sigurnosnih kompanija, specijalizirane vijesti i internetske stranice. S obzirom na to da su ovi izvori dovoljno pouzdani da budu korišteni na vrhunskim konferencijama, kao takvi mogu se smatrati pouzdanim izvorima informacija o računalnoj sigurnosti.

U budućnosti, svakako bi bilo zanimljivo isprobati ovakvu analizu metodama umjetne inteligencije i strojnog učenja, kao što je primjerice obrada prirodnog jezika (engl. *natural language processing*).

# Literatura


[1] Security Magazine. *Global cyberattacks increased 38% in 2022*, (2023, siječanj). https://www.securitymagazine.com/articles/98810-global-cyberattacks-increased-38-in-2022

[2] Microsoft. Microsoft Academic, Conferences Analytics: Computer security, 2021. https://academic.microsoft.com/conferences/41008148,38652104

[3] Dalibor Gernhardt, Stjepan Groš. *Use of a non-peer reviewed sources in cyber-security scientific research.* 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), 2022.

[4] Shan S., Wenger E., Wang B., Bo Li, Zheng H., Y. Zhao B., *Gotta Catch 'Em All: Using Honeypots to Catch Adversarial Attacks on Neural Networks*, ACM SIGSAC Conference on Computer and Communications Security (CCS '20), (2020, studeni).

[5] Yusuke Shinyama, Philippe Guglielmetti & Pieter Marsman. (2022). *pdfminer.six* (v20221105). https://pdfminersix.readthedocs.io/en/latest/

[6] Singer-Vine, J., & The pdfplumber contributors. (2023). *pdfplumber* (Version 0.9.0). https://github.com/jsvine/pdfplumber

[7] Fenniak, M., Stamy, M., pubpub-zz, Thoma, M., Peveler, M., exiledkingcc, & PyPDF Contributors. *PyPDF* (Version 2022). https://pypi.org/project/pypdf/

[8] Duncan Riley and SiliconANGLE Media. 2019. *Symantec is now NortonLifeLock as Broadcom closes purchase of its enterprise business*. (2019, studeni). https://siliconangle.com/2019/11/05/symantec-now-nortonlifelock-broadcom-completes-acquisition-enterprise-business/

[9] Palo Alto Networks. *Palo Alto Networks Completes Acquisition of PureSec*, (2019, lipanj). https://www.paloaltonetworks.com/company/press/2019/palo-alto-networks-completes-acquisition-of-puresec


.

# Analiza korištenja stručnih izvora u znanstvenim radovima iz područja kibernetičke sigurnosti

## Sažetak

Ovaj rad fokusira se na problematiku identifikacije pouzdanih, ali nerecenziranih izvora informacija u znanstvenim radovima iz područja kibernetičke sigurnosti. Kroz analizu referenci u najbolje ocijenjenim znanstvenim radovima s vodećih svjetskih konferencija, razvijena je metodologiju za identificiranje izvora i analizu konteksta u kojem se koriste takvi izvori. Koristeći programski jezik Python, obrađeno je 530 radova, a kao glavni izvori identificirani su publikacije i blogovi sigurnosnih kompanija te specijalizirane i opće vijesti. Rezultati istraživanja pokazuju da se nerecenzirani izvori najčešće koriste za motivaciju rada, odnosno u uvodnom dijelu rada, zatim slijedi razrada teme i glavni dio istraživanja, dok najmanji dio otpada na raspravu i donošenje zaključaka. U radu je zaključeno da su identificirani nerecenzirani izvori, zbog svoje učestale upotrebe na prestižnim konferencijama, pouzdani izvori informacija o kibernetičkoj sigurnosti. Kao budući korak za daljnje analize, predlaže se primjena metoda umjetne inteligencije i strojnog učenja, poput obrade prirodnog jezika.

**Ključne riječi:** računalna sigurnost, nerecenzirani izvori, PDF parser, analiza teksta, regularni izrazi

# Analysis of use of professional sources in cyber security scientific papers

## Summary

This thesis focuses on the issue of identifying reliable, but non-peer reviewed sources of information in cyber-security scientific research. Through a comprehensive examination of references found in top-tier articles presented at the leading conferences, we have developed a methodology for extracting such sources and analyzing the context in which they are utilized. 530 articles were processed using the Python programming language, with the identified sources primarily originating from security companies' publications and blogs, as well as specialized and general news. The findings from the research indicate that non-peer reviewed sources are mostly used as motivation in the introductory sections, followed by the development and main research topics. In contrast, such sources have a less pronounced influence on the discussion and conclusion segments. The study concludes that the identified non-peer reviewed sources, due to their frequent use at prestigious conferences, are reliable sources of information in cyber-security research. For future analyses, the recommendation is to utilize some form of artificial intelligence and machine learning techniques, such as natural language processing.

**Keywords:** cyber security, non-peer reviewed sources, PDF parser, text analysis, regular expressions